USER GUIDE
of
supLeger and subLegerWebJSF
V1.0

Copyright and warning :
This document and the implementation of the light supervisor is free of rights. The user uses this product at his risks. The licence which bracket is licence LGPL. The author declines his responsibility on possible a faulty operation for the product and induced effects

Modifications tracking

Author	*Version*	*Date*	*Modifications*
JL Pasturel	V1.0	30/04/2006	Initialization

Table des matières

1 GOAL OF THE TOOL....4
1.1 Notice preliminary....4
1.2Fast definition of the tool....4
1.2.1 supLeger part :....4
1.2.2SupLegerWebJSF part :....4
1.3Architecture and principles of functioning....4
1.3.1SupLeger part :....4
1.3.2 supLegerWebJSF part :....5
2USER GUIDE supLeger....8
2.1 Reminder of the role of supLeger....8
2.2 Parameters of server part of collect server of supLeger....8
2.3Launching collect server of supLeger....8
2.4Parameters of Agent part of supLeger....8
2.4.1 Trees agent part....9
2.4.2 instances.conf....9
2.4.3 supLegerAgent.conf....10
2.4.4 requetes.conf....10
2.4.5Signification of the fields of the file requetes.conf....11
2.4.6Control of the validity of file requetes.conf....12
2.4.7 Writing request of SHELL type....13
2.4.8 Writing requests of JAVA type....14
2.5 Launching supleger Agent....17
3 USER GUIDE of supLegerWebJSF....18
3.1Reminder of the role of supLegerWebJSF....18
3.2 Launching supLegerWeJSF....18
3.3 Chain of screens for Real Time Supervision....18
3.3.1Connection....18
3.3.2 Choose servers or instances for real Time supervising....21
3.3.3 Choose of refresh frequency....22
3.3.4 Launch Real Time Supervision....24
3.4 Chain of screens of Differed Time supervisor....26
3.5 To Join the author....28
4ANNEXE : Licence LGPL....29

1 GOAL OF THE TOOL

1.1 Notice preliminary

First remark about English document version : English is not my native language, and sorry for the mistakes : .if anybody can correct the most important of them, it would be thanked ! !

The essential purpose during the realization of this tool, was for me to take competence around Java Server Faces. On the other hand, I had already realized this tool with echo / echopoint V1 frameworks (that I did not publish), I thus resumed, of the not published version, the part accesses and managements of the data in the database with Hibernate V2.1. Obviously, the choice of the architecture may be discussed.

1.2Fast definition of the tool

The tool consists of two parts :

1.2.1 supLeger part :

It is a question of realizing a supervisor of functioning of UNIX machine or Windows machine consisted of an agent running on applications or data servers and of a server collating the information of all the servers.

1.2.2SupLegerWebJSF part :

This part is in charge of exploiting these data by realizing the functions of almost real-time supervisor and supervisor batch mode.

Notas :

- in this document we shall propose a fast installation for people been used to the installations
- License of the product: LGPL to see in appendix the contents of the license..

1.3Architecture and principles of functioning

The plan 1, at the end of this paragraph, gives the software complete architecture of both parts of this light supervisor.

1.3.1SupLeger part :

The part to supLeger, is loaded with the data collection indicators of the applications and the systems implanted on servers. The agent and the server of supervision are written in Java. A JVM 1.3 for the least highly-rated agent and a JVM 1.4 in the least highly-rated supervisor is needed.

1 server can welcome n applications. We can define p indicators by application. The agent allows to re-stick these indicators. The collector of supLeger can welcome the data of supervisions of several servers.

- **Agent side :**

Productions of these indicators can be made of 2 manners on every server and application:

- case 1 : writing **SHELL** mode : in Unis shell, Windows batch, C programm, java, perl, Visual Basic ...
- case 2 : writing **JAVA** classe called directly by the agent.

In both cases, it is mandatory that the return function has the pattern :

<return_string_form>;<additionnal_data_string_form>

The **return_string_form** can contain data of type int, float, double, date (with a manadatory pattern) and String. They will be interpreted according to a file of configuration which we shall see farther.

The **additionnal_data_string_form** part allows the return of contextual datas to complete the result.

In the case 1, the return must be done in the stdout (ex function echo for Shell scripts and Windows batches, printf for C function, System.out.println() for Java methods).

In the case 2, the Java class possesses a « retour » method that return a string wihn the pattern seen above.

We shall understand that we have to write once in the **stdout** !

With these returns and configuration file, the Agent builds messages and afterwards sends its to the central supervisor.

Supervisor side :

The supervisor is in charge of receiving messages from all the agents throw a TCP/IP connexion (on a scket) and to load its in the database.

For this, Hibernate O/R mapping is used. From fine grained business objects, Hibernate maps this objets with database tables. The database is very simple : only 10 tables (including relation tables). The data engine is PostgeSQL V8, that is now available in native mode on Windows OS and Linux.

Number of POJOs is 8.

1.3.2 supLegerWebJSF part :

It is the visible part for the user. It allows to realize the display of indicators is in continuous mode (function of supervision « almost » real time) or in deferred mode and over a customizable period by the user (function supervision deferred mode).

This part runs with the datas agregated by the supLeger part..
The engine of Web Server application used is Winstone. It is about a light container of servlet / JSP (under double license LGPL and CDDL) that we can find at this address:

http://winstone.sourceforge.net/

I have to thank the author (**Rick Knowles**) who helped me a lot to integrate my tool into his server (that he had in priori not designed to run with JSF, and Rick helped me to realize this integration).

In the paragraph 4 of Install manual, I indicate how to integrate my tool into other Servlet / JSP containers based on Tomcat (ex JONAS).

For the display of the data, we used a framework: : JSF (Server Faces Java) and its Apache implementation MyFaces version 1.1.1
This framework is under Apache license V2.0.

MyFaces : http://myfaces.apache.org/

To uncouple the part View by the model part, the aspect business and manipulation of the data is delegated to objects facades coarse-grained. For this project, I kept adhesions between the View part and the business to insure a better performance (I did not use a pattern Dated Transfer Object.). That is that to use facades in another context of display (Fat clients, Struts), it would be necessary to retouch slightly in the code of facades.
The management of the access to the data and the persistence (read only in fact) also crosses by the mechanism POJOs / Hibernate and database PostgreSQL.
Noted: it is the same pojos that those of the part to supLeger, but the code is replicated because we can have architectures where the supervisor of supLeger is not on the same machine as the WAS (Winstone) of subLegerWeb. For the same reason the framework Hibernate is duplicated in supLeger and supLegerWeb. See Install manual, for installations possibities.

Plan 1: software Architecture

The schema below shows all the components of supLeger and supLegerWebJSF :

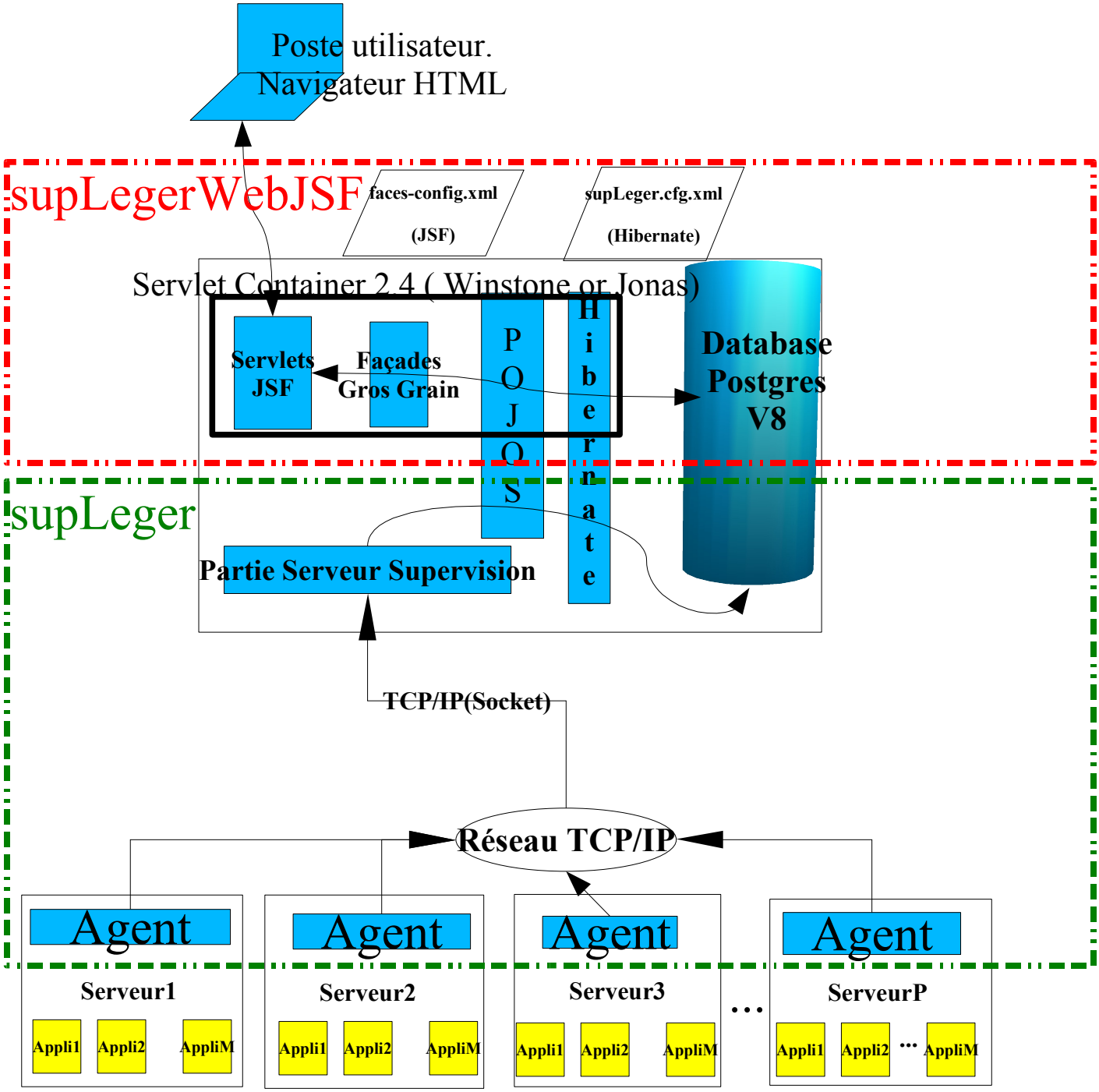


Schéma 1 : Software Architecture

2USER GUIDE supLeger

2.1 Reminder of the role of supLeger

supLeger is in charge of collecting datas from different agents located on servers and to load them in the database PostgreSQL. The **Agents** are part of **supLeger**.

Report on Install Manual to install the Agents on servers that will be supervised..

2.2 Parameters of server part of collect server of supLeger

Under directory <supLeger_root>\super\conf, there is only one configuration file : **supLegerServeur.conf** taht indicate socket number that listens :

.

supLegerServeur.conf :

```
SOCKET=5712
```

and it is all ! ...

2.3Launching collect server of supLeger

To launch this server, from a system console, type : **lanceServeurThreadW.bat** (Windows) or **lanceServeurThreadL.ksh** (*nix)

2.4Parameters of Agent part of supLeger

It is the part located on server s that we want to supervise the applications that run on these servers.

2.4.1 Trees agent part

The table below show this trees :

<AGENT_ROOT>	*First level*	*Second Level*	*Third Level*	*Comments*
conf (rep)	**instances.conf** (fic)			Lists of applications to supervise
	supLegerAgent.conf (fic)			Network Configuration of the agent
	requetes.conf (fic)			Parameters of requests of the agent
requetes (rep)	**SHELL (rep)**	**all** (rep)	**all_requete1.bat** (fic) **all_requete2.bat ...**(fic)	Shell requests for all applications of the server (.bat = Win, .ksh = *nix)
		monAppli1 (rep)	**a1_requete.bat ...**(fic)	Shell requests for the application monAppli1.
		monAppli2 (rep)...	**a2_requete.bat ...**(fic)	Shell requests for the application monAppli2
	JAVA (rep)	**all** (rep)	**All_requete1.class ...**(fic)	Classes Java or all applications of the server
		monAppli1(rep)	**A1_requete1.class ...**(fic)	Classes Java for the application monAppli1.
		monAppli2 (rep)...	**A2_requete1.class ...**(fic)	Classes Java for the application monAppli2.
lanceAgentW.bat or **lanceAgentL.ksh** (fic)				File for launching the agent
supLeger-agent.jar (fic)				Archive of java binaries of the agent

Under the directory **<supLeger_root>\agent\conf**, the configuration files are :

- **instances.conf**
- **supLegerAgent.conf**
- **requetes.conf**

2.4.2 instances.conf

The file contains only one line, that lists the applications that we want to supervise. Names of the application are separated by a colon

Sample of File **instances.conf** :

```
instances = monAppli1;monAppli2
```

2.4.3 supLegerAgent.conf

This file determine :

- the hostname of the Agent server. This name is free, it is not mandatory to give the real hostname.
- Address (DNS or IP address) of the central server that collects datas and socket port of central server
- socket port for the agent side.

Below, an example of **supLegerAgent.conf :**

```
#IP_ADRESSE=127.0.0.1:5712
HOSTNAME=localhost
DNS_ADRESSE=localhost.localdomain:5712
SOCKET_LOCAL=5306
```

Nota : port **5712** corresponds to listen port of central server (Parameter SOCKET of **supLegerServeur.conf**)

2.4.4 requetes.conf

It is the file that list the requests and its parameters. The parameters are frequency, applications, nominal result, warning, alarm, rules. Alls this parameters are detailed in a table more far.

The file consists of lines, each line consists on fields separated by colon (CSV format)

Example of file **requêtes.conf** :

```
#<INSTANCE>;<REQUEST_NAME>;<TYPE_ANALYZE:/INT/LONG/FLOAT/STRING/DATE>;<TYPE :
SHELL/JAVA>;<FREQUENCY_LAUNCHING_IN_S><NOMINAL_VALUE>;<WARNING_VALUE>;<ALARM_VALUE>;<RULE>;<PARA
METERS>;
# All the attributes must be set. We must set NONE to an attribute without signification
# In last field, several parameters can be set, they must be separated by the & character
#
#all;presenceProcessusJavaw;DOUBLE;SHELL;w_presenceProcessus.bat;10;1;2;0;EGAL;javaw;
#monAppli1;presenceProcessusTutu1;DOUBLE;SHELL;w_presenceProcessus.ksh;15;0.98;1.02;0;SUP;java;
#monAppli1;presenceProcessusTutu2;DOUBLE;SHELL;w_presenceProcessus.ksh;10;0.98;1.02;0;SUP;java;
#monAppli1;presenceProcessusTutu3;DOUBLE;SHELL;w_presenceProcessus.ksh;20;0.98;1.02;0;SUP;java;
#monAppli1;presenceProcessusTutu4;DOUBLE;SHELL;w_presenceProcessus.ksh;12;3;4;0;SUP;java;
#monAppli1;presenceProcessusTutu5;DOUBLE;SHELL;w_presenceProcessus.ksh;15;0.98;1.02;0;SUP;java;
#monAppli1;presenceProcessusTutu6;DOUBLE;SHELL;w_presenceProcessus.ksh;10;0.98;1.02;0;SUP;java;
#monAppli1;presenceProcessusTutu7;DOUBLE;SHELL;w_presenceProcessus.ksh;20;0.98;1.02;0;SUP;java;
#monAppli1;presenceProcessusTutu8;DOUBLE;SHELL;w_presenceProcessus.ksh;12;3;4;0;SUP;java;
#monAppli1;presenceProcessusTutu9;DOUBLE;SHELL;w_presenceProcessus.ksh;15;0.98;1.02;0;SUP;java;
#monAppli1;presenceProcessusTutu10;DOUBLE;SHELL;w_presenceProcessus.ksh;10;0.98;1.02;0;SUP;java;
#monAppli1;presenceProcessusTutu11;DOUBLE;SHELL;w_presenceProcessus.ksh;20;0.98;1.02;0;SUP;java;
#monAppli1;presenceProcessusTutu12;DOUBLE;SHELL;w_presenceProcessus.ksh;12;3;4;0;SUP;java;
all;presenceProcessusTutu1;DOUBLE;SHELL;w_presenceProcessus.bat;15;0.98;1.02;0;BETWEEN;java;
monAppli1;presenceProcessusTutu2;FLOAT;SHELL;w_presenceProcessus.bat;10;0;1;1;EGAL;java1;
monAppli1;presenceProcessusTutu3;FLOAT;SHELL;w_presenceProcessus.bat;11;1;0;0;EGAL;java2;
monAppli1;presenceProcessusTutu4;FLOAT;SHELL;w_presenceProcessus.bat;12;1;0;0;EGAL;java3;
monAppli1;presenceProcessusTutu5;FLOAT;SHELL;w_presenceProcessus.bat;13;1;0;0;EGAL;java4;
monAppli1;presenceProcessusTutu6;FLOAT;SHELL;w_presenceProcessus.bat;14;1;0;0;EGAL;java5;
monAppli1;presenceProcessusTutu7;FLOAT;SHELL;w_presenceProcessus.bat;15;1;0;0;EGAL;java6;
monAppli1;presenceProcessusTutu8;FLOAT;SHELL;w_presenceProcessus.bat;16;1;0;0;EGAL;java7;
monAppli1;presenceProcessusTutu9;FLOAT;SHELL;w_presenceProcessus.bat;17;1;0;0;EGAL;java8;
monAppli1;presenceProcessusTutu10;FLOAT;SHELL;w_presenceProcessus.bat;18;1;0;0;EGAL;java9;
```

```
monAppli1;presenceProcessusTutu11;FLOAT;SHELL;w_presenceProcessus.bat;19;1;0;0;EGAL;java10;
monAppli1;presenceProcessusTutu12;FLOAT;SHELL;w_presenceProcessus.bat;10;1;0;0;EGAL;javaw1;
#monAppli2;presence3Processus;DOUBLE;SHELL;w_presenceProcessus.bat;21;3.01;2;1;SUP_EGAL;java12;
monAppli2;dateFic1;DATE;SHELL;w_dateFic1.bat;22;-210000;-
250000;+0;SUP_EGAL;C:\opt\supLeger\agent\requetes\SHELL\monAppli2&w_dateFic1.bat;
monAppli2;dateFic2;DATE;JAVA;DateFic1;23;-
30000;+60;+30000;BETWEEN;C:\opt\supLeger\agent\requetes\JAVA\monAppli2&DateFic1.java;
monAppli2;dateFic2Relative;DATE_RELATIVE;JAVA;DateFic1;23;-
30000;+60;+30000;BETWEEN;C:\opt\supLeger\agent\requetes\JAVA\monAppli2&DateFic1.java;
monAppli2;dateFic2Absolue;DATE_ABSOLUTE;JAVA;DateFic1;23;2006/03/10 00:00:00;2006/03/20 00:00:00;2006/03/30
00:00:00;BETWEEN;C:\opt\supLeger\agent\requetes\JAVA\monAppli2&DateFic1.java;
monAppli2;PresFic1;DOUBLE;JAVA;PresFic1;24;0.98;1.02;0;BETWEEN;C:\opt\supLeger\agent\requetes\JAVA\monAppli2&PresFic1.java;
```

2.4.5Signification of the fields of the file requetes.conf

The table below, summarize all the fields of the file **requetes.conf** :

No	*Name of the field*	*Possible values*	*Signification / Comments*
1	INSTANCE	String	Name of the application on which the request is launched. The reserved word “all” indicates that the request concerns all the application of the server (listed in **instances.conf**). **Attention ! this name must be unique on the central database.**
2	REQUEST_NAME	String	Name of the request. **Attention ! this name must be unique on the central database.**
3	ANALYSE_TYPE	INT ; DOUBLE ; FLOAT ; LONG ; DATE_RELATIVE;DATE_ABSOLUTE; STRING	Indicate the java type of the result on which the analyse will be done.
4	REQUEST_TYPE	SHELL ; JAVA	Indicate that the request is a java request or a shell request. In case of Java request, we use the same JVM taht tg=he agent, and it improves performance.
5	EXECUTABLE	String	It must be corresponding to the name of the program or the Class Java that must be executed . **These programs or Java Classes must be located corresponding to field 1 and 4.** D
6	FREQUENCY	Int	Indicate frequency in seconds of the request
7	NOMINAL_VALUE	Depends on type	For STRING result, that can give multiples results, separate STRING with a pipe (|)
8	WARNING_VALUE	Depends on type	For STRING result, that can give multiples results, separate STRING with a pipe (|)
9	ALARM8VALUE	Depends on type	For STRING result, that can give multiples results, separate STRING with a pipe (|)
10	RULE	EGAL ; INF; SUP; BETWEEN; DIFF;	Rules that permit to qualify the result : Correct : Green fire Warning : Orange fire Alarm : Red fire in supLegerWebJSF. For STRING in version 1.0, we manage only the EGAL anfd DIFF rules.
11	PARAMETETS	String characters separated by &	Permit to pass parameters to programm (Shell or java classes)

Particular case of the treatments of dates:

a) DATE_RELATIVE

Nota : DATE_RELATIVE has the same meaning than DATE

It is when ANALYSE_TYPE is equals to DATE_RELATIVE and how to fill NOMINAL_VALUE , WARNING_VALUE and ALARM_VALUE (fields 7,8,9).

Value of these fields are measured by the gap in seconds with regard to the recording date of the measurement.

For example a value like -20000 (in field 7 or 8 or 9) , indicates that the date is the date of the recording minus 20000 seconds.

b) DATE_ABSOLUTE

It is the case when we want to compare dates with the mandatory pattern below :

YYYY/MM/DD HH:mm:ss

an error in the format raises an exception in the supLegerWebJSF browser.

2.4.6Control of the validity of file requetes.conf

When the agent starts, controls of requetes.conf are made :

- number of fields per lines
- date format in DATE-RELATIVE mode and DATE_ABSOLUTE mode

The output is sent to the console.

We can also verify requetes.conf under the directory :

<supLeger_root>/agent

launch :

java -classpath ./supLeger-agent.jar agent.scheduleur.VerifConfRequetes

2.4.7 Writing request of SHELL type

The structure of the result must be like the pattern below :

<Result_String>;<Extra_datas_String>

In case of batch .bat, .cmd or ksh, it is sufficient that we return result in an **echo** function or by the **type** function on a file (the result must be sent to the stdout)..

The parameter of the shell are described in the file **requêtes.conf** seen before**.**

We must copy the request on the directory according the description of the request in the file **requêtes.conf (see fields 1 and 4)**

Attention, it is mandatory to have only one wrinting in the stdout !!.

Example of request wrote with **msys on Windows**, verifying date of a file w_dateFic1.bat :

```
@echo off
REM Compter les occurences d un processus
set PATHOLD=%PATH%
set MSYS_BIN=c:\msys\1.0\bin
set AGENT_ROOT=c:\opt\supLeger\agent
set PATH=%MSYS_BIN%;%PATH%
cd %1
dir  %2 |grep %2 | cut -d " " -f1 |cut -d "/" -f1 > JJ
dir  %2 |grep %2 |cut -d " " -f1 | cut -d "/" -f2 > MM
dir  %2 |grep %2 |cut -d " " -f1 | cut -d "/" -f3 > YYYY
dir %2| grep %2 |cut -d " " -f3 | cut -d ":" -f1 > HH
dir  %2 |grep %2 | cut -d " " -f3 | cut -d ":" -f2 > min

type YYYY>jlp3
echo />> jlp3
type MM>>jlp3
echo />> jlp3
type JJ>>jlp3
echo + >> jlp3
type  HH>>jlp3
echo :>>jlp3
type min>> jlp3
echo :00>> jlp3
type jlp3 | tr -d "\r" | tr -d "\n" |tr -d "+" > jlp4
echo ;%2>> jlp4
type jlp4


del jlp3
del jlp4
del JJ MM YYYY HH min
cd %AGENT_ROOT%
set PATH=%PATHOLD%
```

It is more simple in Unix ksh !

Another example to detect processus (msys mandatory) :

```
@echo off
set PATHOLD=%PATH%
set MSYS_BIN=c:\msys\1.0\bin
set AGENT_ROOT=c:\opt\supLeger\agent
set PATH=%MSYS_BIN%;%PATH%
ps -W | grep %1 | wc -l | tr -d "\n" | tr -d "\r" && echo ;%1
set PATH=%PATHOLD%
cd %AGENT_ROOT%
```

2.4.8 Writing requests of JAVA type

Rules to write a request in Java for supLeger :

Règles d'écriture de la classe Java :

- to simplify its writing, the Java Class will **not** belong to a package.
- The Java Class possess a function **retour** (the name "retour" is mandatory) with the prototype below :

public String retour(String[] args)
The array args contains the parameters of the last field of thecorresponding line in requetes.conf

The return of this function will be in format String with the pattern beloww :
<Result_String>;<Extra_datas_String>

To compile, put the .java file, you can compile it on any computer by command :

javac -classpath . MaRequete.java

and afterwargs put the .class file, on the server agent , in the good directory (according the array of § 2.4.1) for example :**<AGENT_ROOT>\requetes\JAVA\monAppli2**

Example of request written in Java Type, verifying the date of a file .

```
import java.io.*;
import java.util.*;

/**
 * @author Jean-Louis
 *
•    Exemple de classe Java chargée depuis la JVM de supLeger
•    Noter le prototype de la fonction retour qu il faut respecter
 */
public class DateFic1 {

        /**
         *
         */
        public String ret;
```

```
	public String fic="";
	public String dir="";
	public DateFic1()
	{}

	public  String retour(String[] args)
	{
		fic=args[1];
		dir=args[0];
		 File f=new File(dir,fic);
		long dateMillis=f.lastModified();
		Date date=new Date(dateMillis);
		Calendar cal=Calendar.getInstance();
		cal.setTime(date);
		String YYYY=Integer.toString(cal.get(Calendar.YEAR));
		System.out.println("DateFic1 : 0");
		String MM=Integer.toString((cal.get(Calendar.MONTH)+1)+100).substring(1);
		System.out.println("DateFic1 : 1");
		String JJ=Integer.toString((cal.get(Calendar.DAY_OF_MONTH))+100).substring(1);

		String HH=Integer.toString(cal.get(Calendar.HOUR_OF_DAY)+100).substring(1);
		String mm=Integer.toString(cal.get(Calendar.MINUTE)+100).substring(1);
		String ss=Integer.toString(cal.get(Calendar.SECOND)+100).substring(1);

		ret=YYYY+"/"+MM+"/"+JJ+" "+HH+":"+mm+":"+ss+";"+fic;
	   return ret;
	}

}
```

Below, another Java request, that has in charge to supervise the number of rows of the table “messagerecu” of supLeger supervisor itself :

```
import java.sql.DriverManager;
import java.sql.Connection;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;

public class CountMessages {

	public CountMessages() {
		super();
		// TODO Auto-generated constructor stub
	}

	public String retour(String[] args) {
		String retour = "";
		try {
			Class.forName("org.postgresql.Driver");
		} catch (ClassNotFoundException cnfe) {
			System.err.println("Couldn't find driver class:");
			cnfe.printStackTrace();
		}
		Connection c = null;
		try {
```

```
                        // The second and third arguments are the username and password,
                        // respectively. They should be whatever is necessary to connect
                        // to the database.
                        c = DriverManager.getConnection(
                                        "jdbc:postgresql://localhost:5432/essai", "essai",
                                        "essai");
                } catch (SQLException se) {
                        System.out
                                        .println("Couldn't connect: print out a stack trace and exit.");
                        se.printStackTrace();
                        System.exit(1);
                }
                Statement s = null;
                try {
                  s = c.createStatement();
                } catch (SQLException se) {
                  System.out.println("We got an exception while creating a statement:" +
                                "that probably means we're no longer connected.");
                  se.printStackTrace();
                  System.exit(1);
                }
                ResultSet rs = null;
                try {
                  rs = s.executeQuery("SELECT count(*)  FROM messagerecu");
                } catch (SQLException se) {
                  System.out.println("We got an exception while executing our query:" +
                                "that probably means our SQL is invalid");
                  se.printStackTrace();
                  System.exit(1);
                }
                int number = 0;
                try {
                        rs.next();
                        number =rs.getInt(1);
                } catch (SQLException e) {
                        // TODO Auto-generated catch block
                        e.printStackTrace();
                }
                retour=Integer.toString(number)+";"+Integer.toString(number)+ " dans table messagerecu";
                c=null;
                return retour;

        }

        /**
         * @param args
         */
        public static void main(String[] args) {
                // TODO Auto-generated method stub
                System.out.println(" retour = " + new CountMessages().retour(args));
        }

}
```

The method main, is only use to test the program. For it runs, we must set the classpath to see the JDBC driver

of Postgres when agent is launched.. The packaging supplied, includes JDBC driver for Postgres Version 8.1, and the classpath is correctly set. If you change Postgres version, you have to modify the classpath in the script that launches the agent.

2.5 Launching supleger Agent

We launch the Agent, by executing in a console system the files : **lanceAgentW.bat** (Windows) or **lanceAgentL.ksh** (*nix) . Under unix the Agent can be launched in background and nohup modes.

3 USER GUIDE of supLegerWebJSF

3.1Reminder of the role of supLegerWebJSF

SupLegerWebJSF allows the visualization of the datas collected by **supLeger.**

These results are stored in a P**ostgreSQL** database.

SupLegerWebJSF permit both quite Real Time visualization and differed visualization of the functioning of applications and servers**.**

SupLegerWebJSF permit also to manage users, servers, applications and requests. The screens are classical CRUF (Creation, Remove, Update, Find) , there are not detailed here because they are trivial.

See Install manual to install **supLegerWebJSF.**

When istallation is finished, there is no configuration to do on the **supLegerWebJSF server.**

3.2 Launching supLegerWeJSF

See if supLeger (Central server and at least one agent) is launched to receive datas in real time.

To launch supLegerWeb, it is enough to launch Winstone in the directory according to it installation : c:\opt\winstone

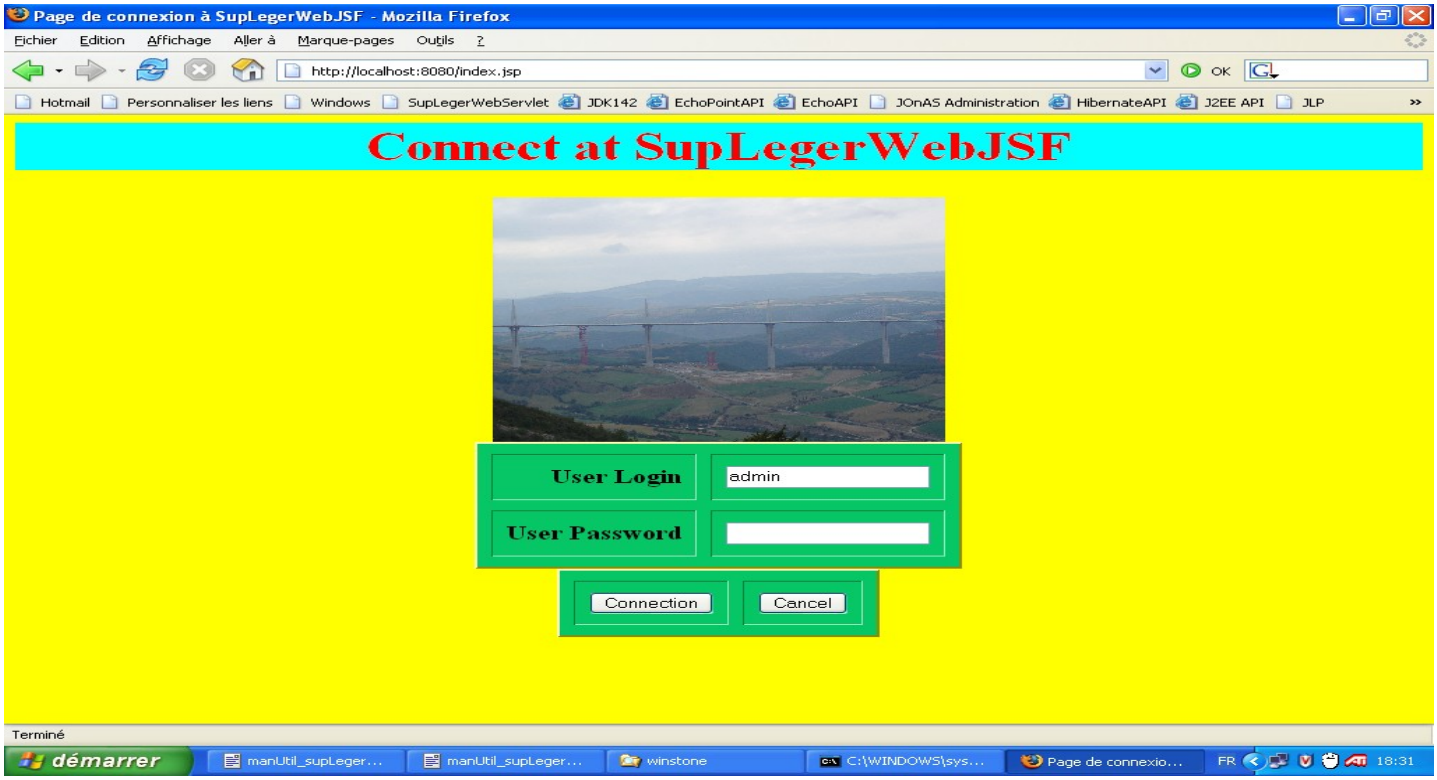
launch :

lanceW.bat or **lanceL.ksh** (as we are ons Windows or on Linux/Unix).

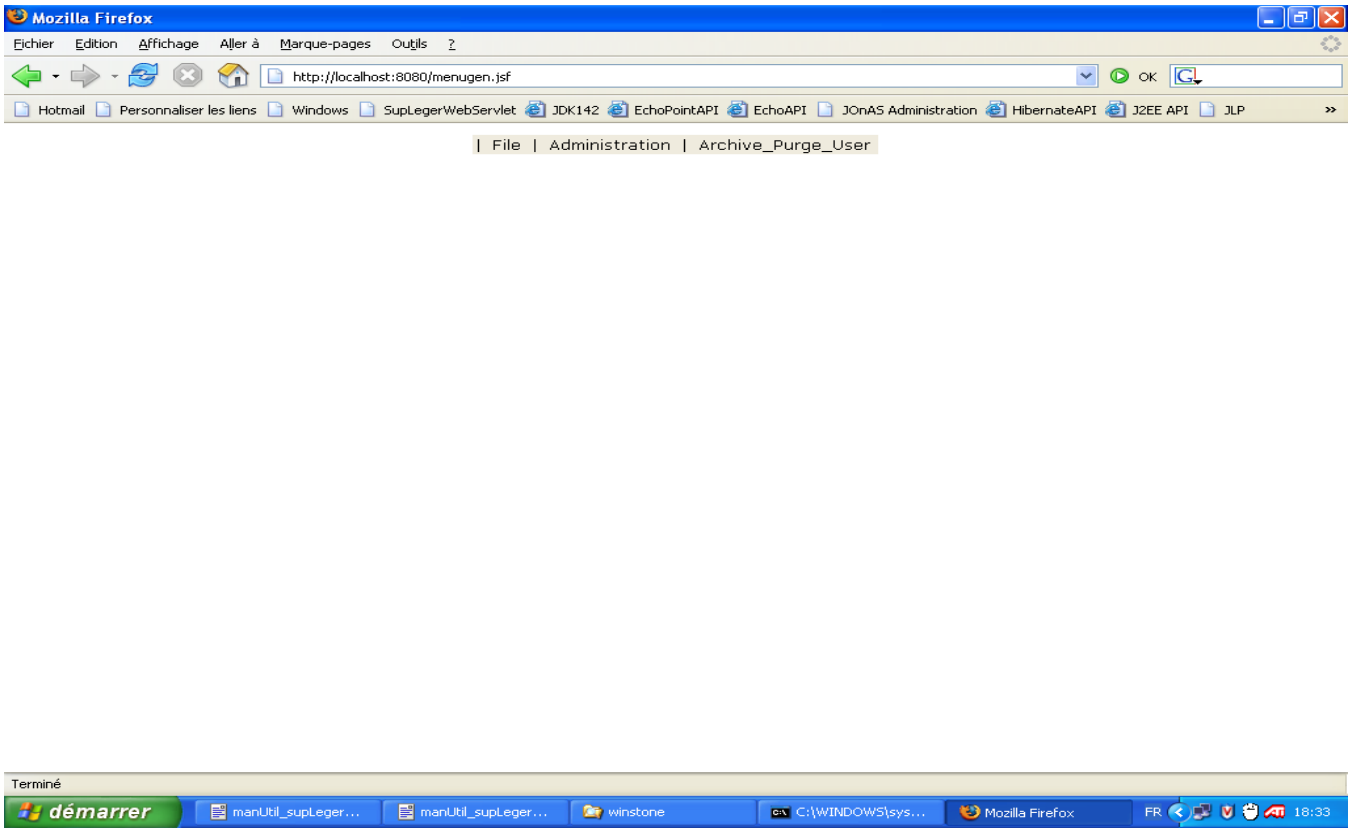
On unix/linux You can launch it in background and nohup modes.

3.3 Chain of screens for Real Time Supervision

3.3.1Connection



Connection to do with login/password : **admin/admin .** Click on **Connection**



Note : If the user has not the **ADMIN** role, the menus are a little bit different : it is not possible to create certain object like users, servers, instances of applications. It is also impossible to create links between servers and users. Purges and archives are possible only on the messages of the user connected, if he is not a user with the **ADMIN** role. **ADMIN** role has more extended possibilities. User will discover by himself all trivial functionalities that are not described in this document.

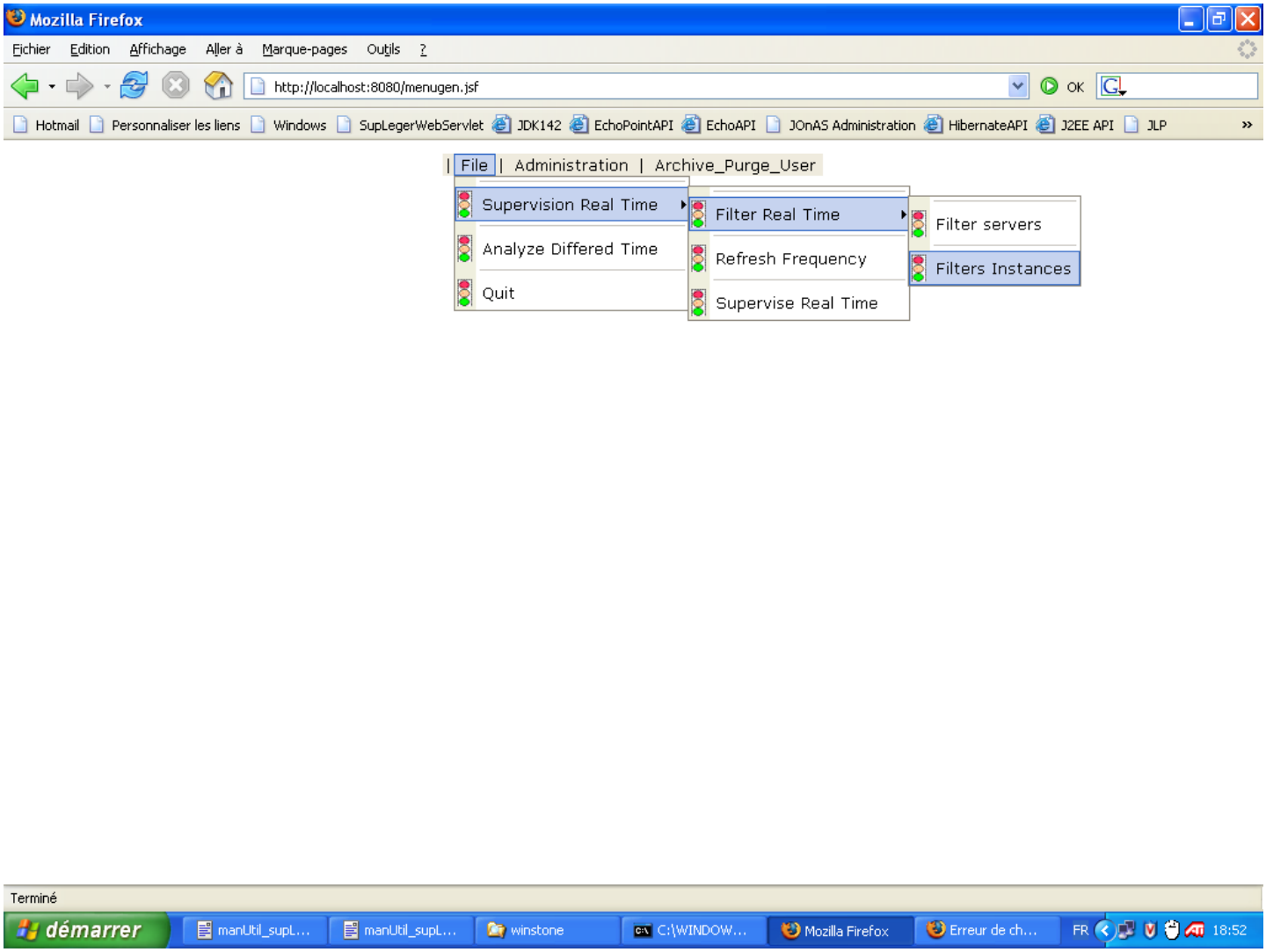
Particular cases of Request screens :

It is about storing in database, the requests that are launched in all the agents. The name field of the request is automatically filled at the start of the agent, if it does not already exist.

On the other hand, it is necessary to fill the source field and the comment field. So database, is used are a referential for requests, This requests can be used on several agents.

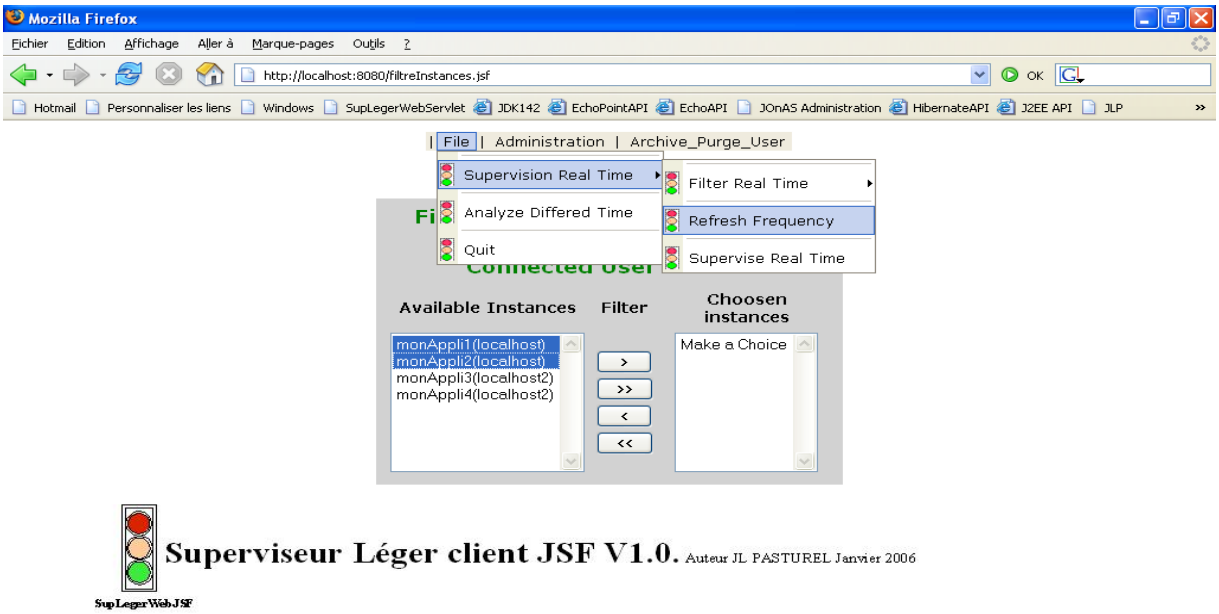
There is no mechanism to deploy requests on different agents.

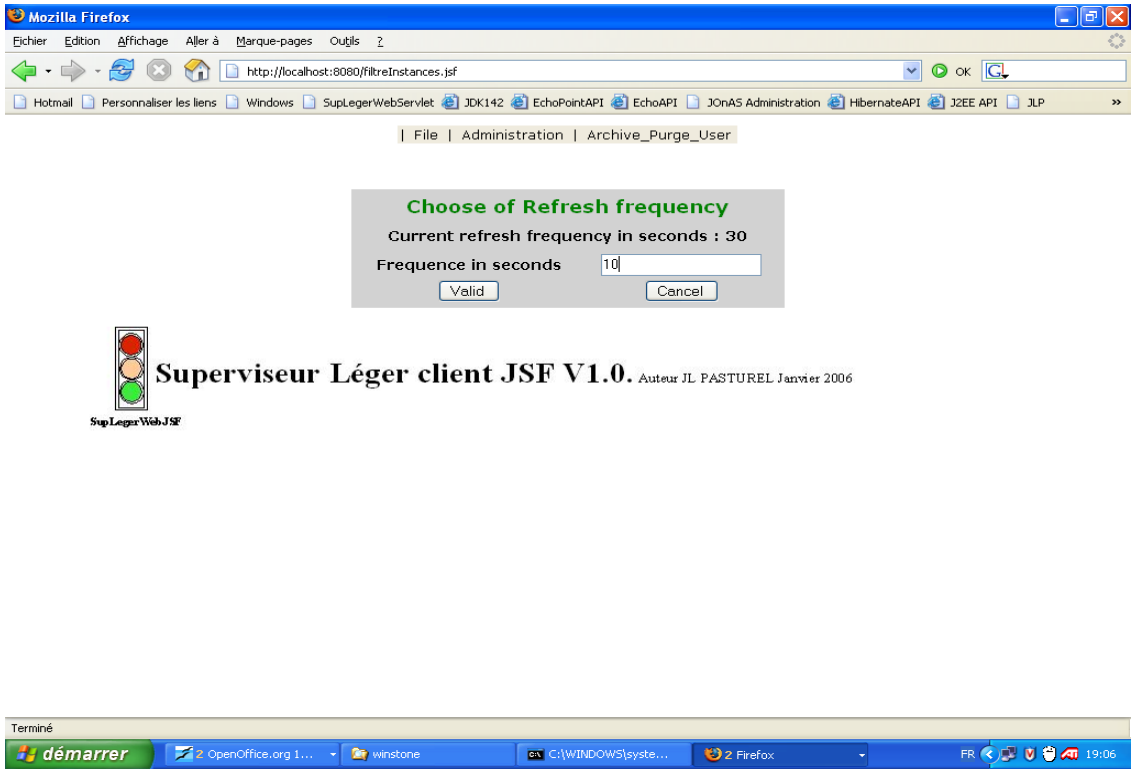
3.3.2 Choose servers or instances for real Time supervising



Choose one or n instances (or servers for sub-menu server) and click to move them to the right side of the chooser.

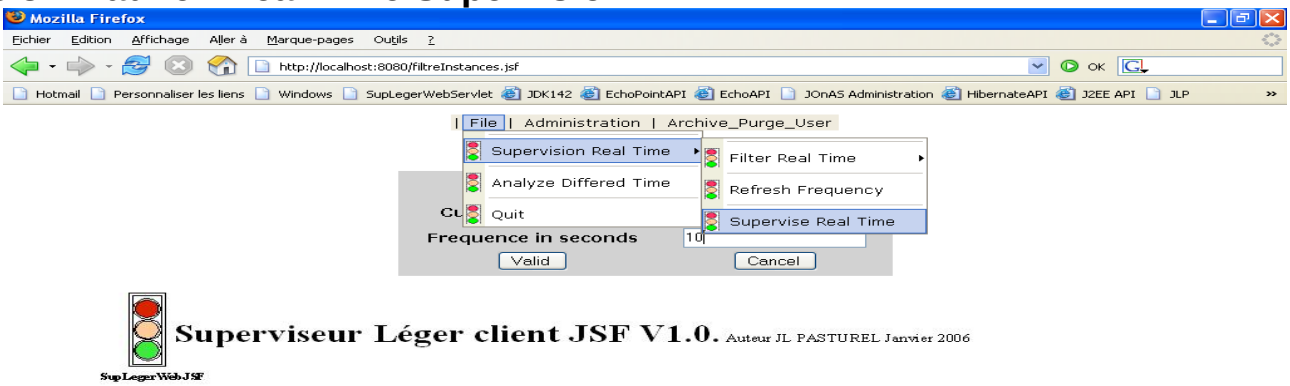
3.3.3 Choose of refresh frequency

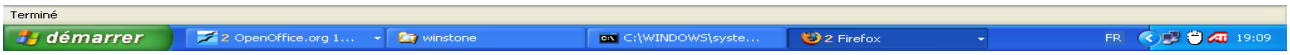


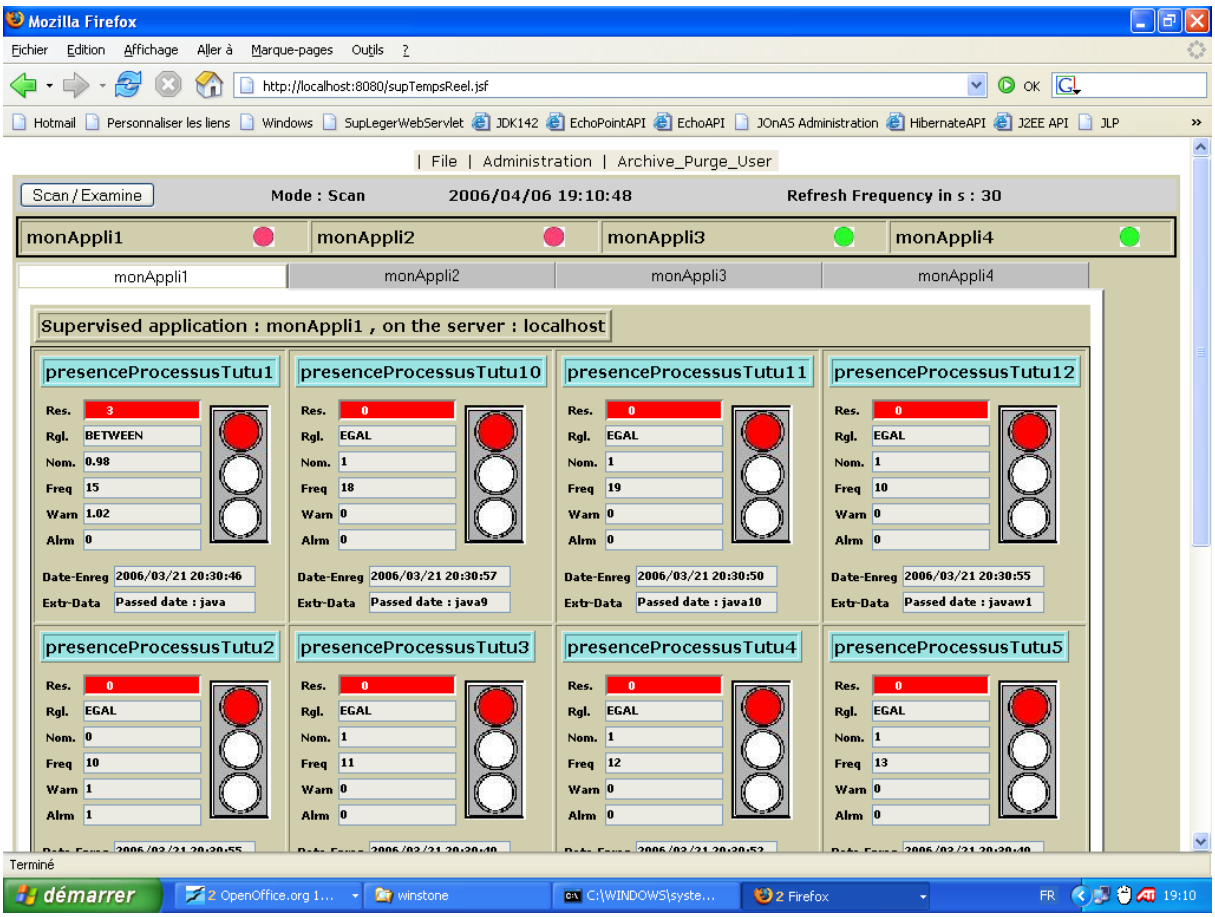


Type frequency in second and click on **Valid**

3.3.4 Launch Real Time Supervision

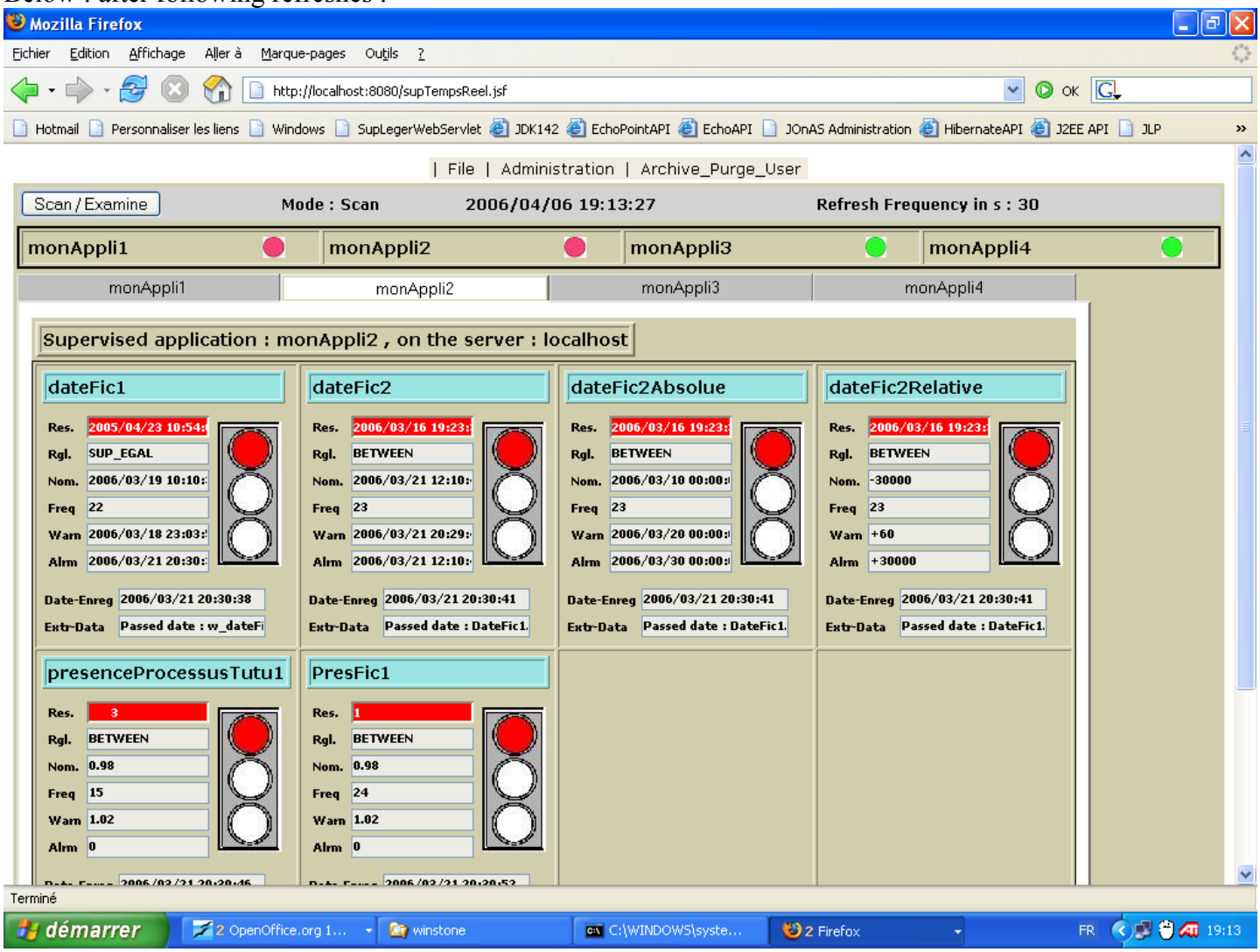




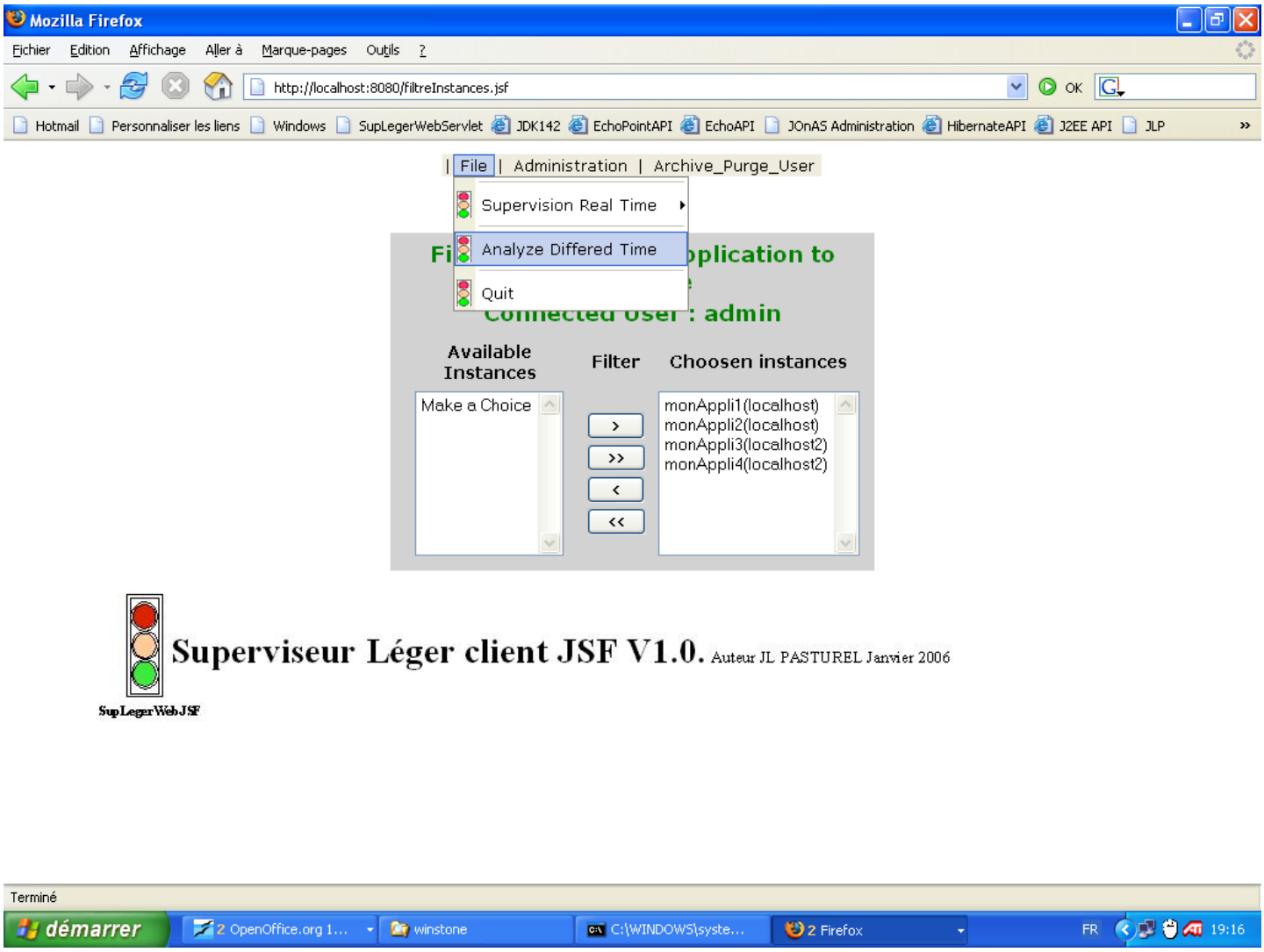


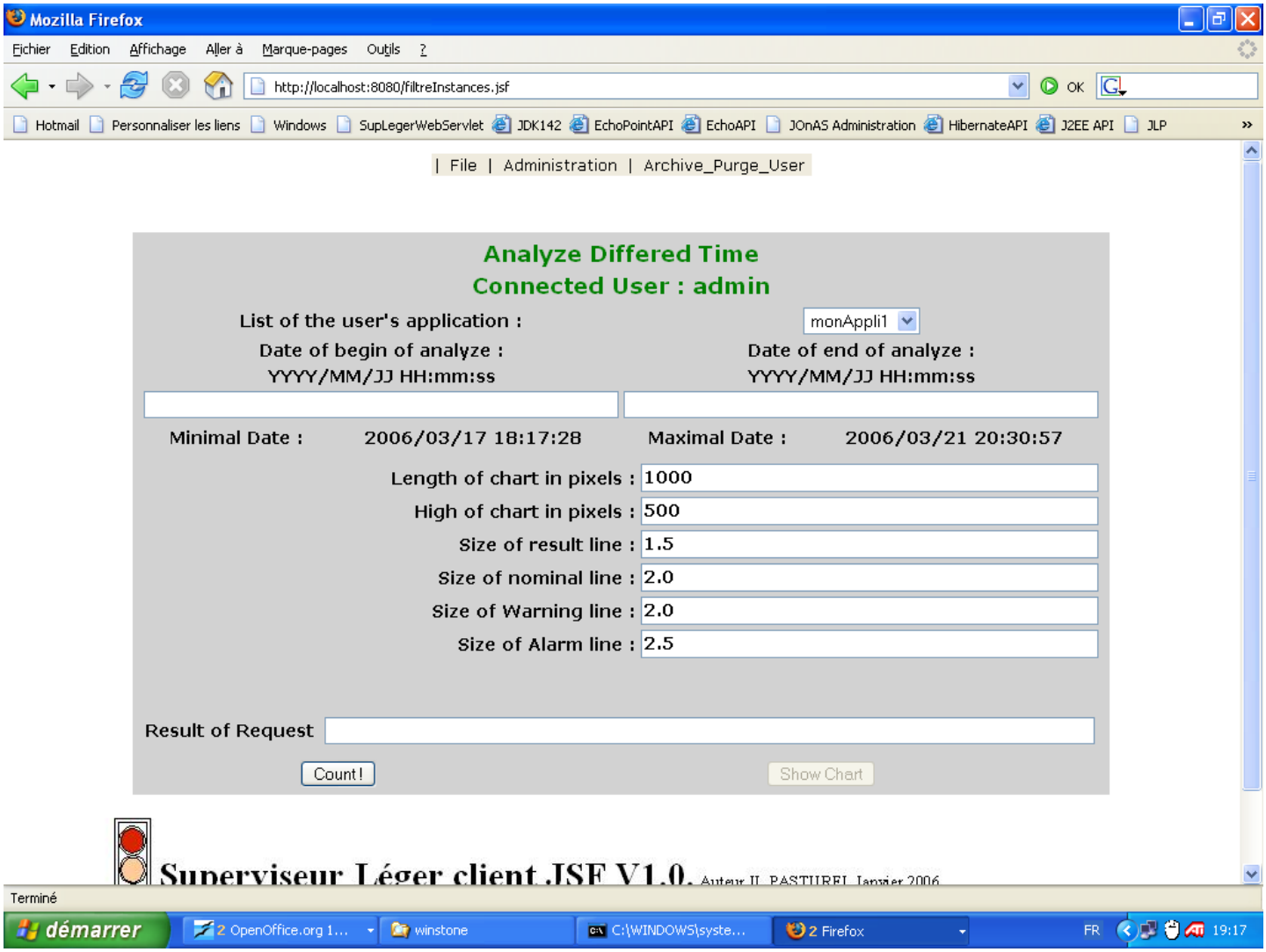
Above : before the first refresh

Below : after following refreshes :



3.4 Chain of screens of Differed Time supervisor




Mozilla Firefox
Fichier Edition Affichage Aller à Marque-pages Outils ?
http://localhost:8080/filtreInstances.jsf
Hotmail Personnaliser les liens Windows SupLegerWebServlet JDK142 EchoPointAPI EchoAPI JOnAS Administration HibernateAPI J2EE API JLP
| File | Administration | Archive_Purge_User
Analyze Differed Time
Connected User : admin
List of the user's application :
monAppli1
Date of begin of analyze :
YYYY/MM/JJ HH:mm:ss
Date of end of analyze :
YYYY/MM/JJ HH:mm:ss
Minimal Date : 2006/03/17 18:17:28
Maximal Date : 2006/03/21 20:30:57
Length of chart in pixels : 1000
High of chart in pixels : 500
Size of result line : 1.5
Size of nominal line : 2.0
Size of Warning line : 2.0
Size of Alarm line : 2.5
Result of Request
Count !
Show Chart
Superviseur Léger client JSF V1.0. Auteur JL PASTUREL Janvier 2006
Terminé
démarrer
2 OpenOffice.org 1...
winstone
C:\WINDOWS\syste...
2 Firefox
FR
19:17

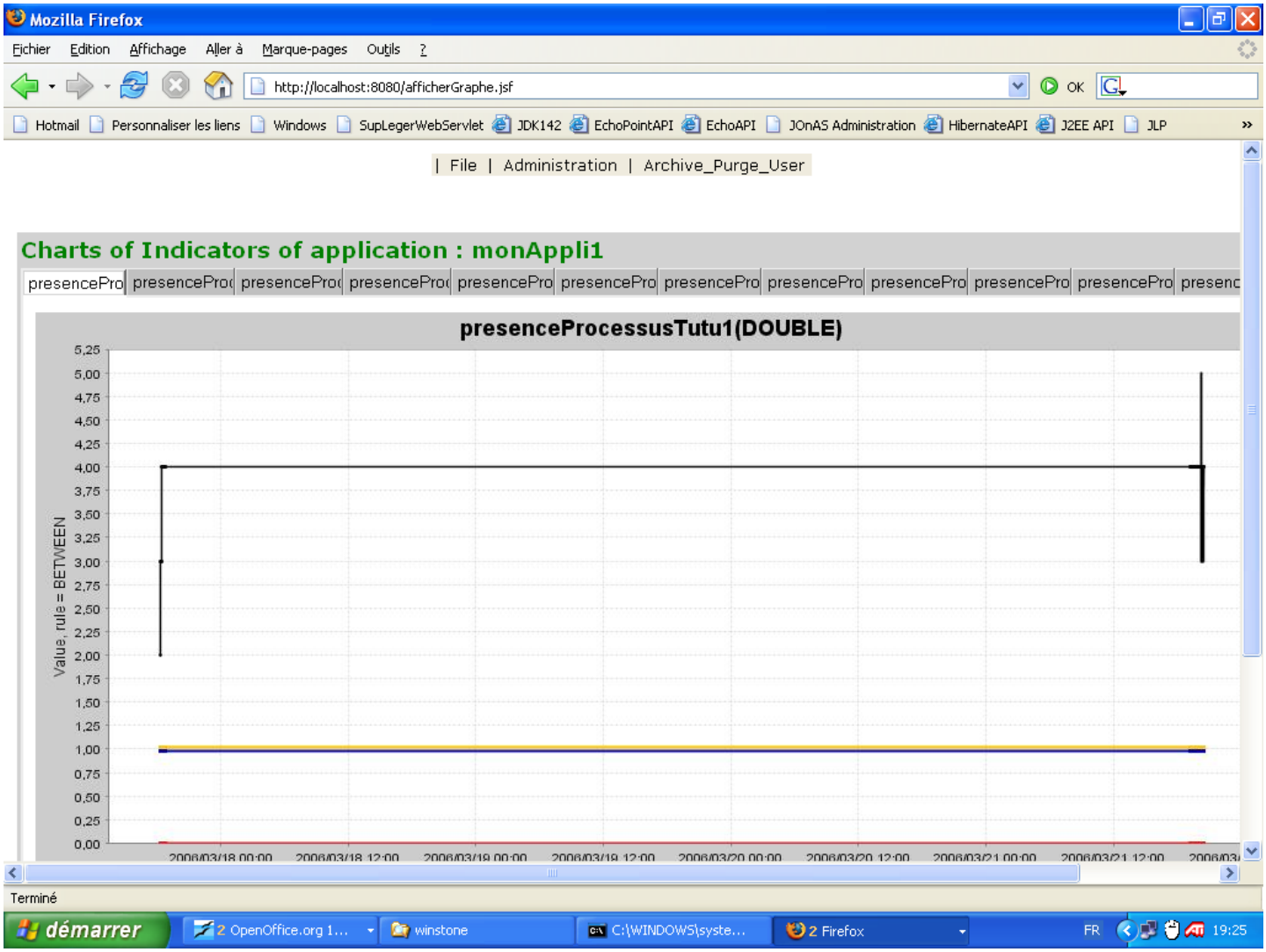
In this screen, we can choose in the combo-bos, instance of application that we want to analyse.

We can define the date of the beginning and the date of the end of the analyse, according to mandatory format.

The others fields permit to parameter the aspect of the chart (size, thickness of the lines)

Once, we have filled all these fields click on **Count!** and afterwards click on **Show Charts**

Chaque onglet affiche, un indicateur différent pour l'instance choisie.



It does not more remain than to write the scripts or Java classes.

Good use of supLeger and supLegerWeb.!!

Enjoy it ..

3.5 To Join the author

For all problems or bugs, you can join me at this e-mail address :

jean-louis.pasturel@wanadoo.fr

.

4ANNEXE : Licence LGPL

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence
the version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public
Licenses are intended to guarantee your freedom to share and change
free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You
can use it too, but we suggest you first think carefully about whether
this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use,
not price. Our General Public Licenses are designed to make sure that
you have the freedom to distribute copies of free software (and charge
for this service if you wish); that you receive source code or can get
it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these
rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source
code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling
it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the
library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know
that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser
General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use
this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using
a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with
the library.

We call this license the "Lesser" General Public License because it
does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less
of an advantage over competing non-free programs. These disadvantages
are the reason we use the ordinary General Public License for many
libraries. However, the Lesser license provides advantages in certain

special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that,
in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has
a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in
themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you
distribute the same sections as part of a whole which is a work based
on the Library, the distribution of the whole must be on the terms of
this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote
it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library
with the Library (or with a work based on the Library) on a volume of
a storage or distribution medium does not bring the other work under
the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public
License instead of this License to a given copy of the Library. To do
this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2,
instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify
that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all
subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany
it with the complete corresponding machine-readable source code, which
must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the
source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or
linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file
that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the
Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6.
Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or
link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the
Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one
of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception,
the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major
components (compiler, kernel, and so on) of the operating system on
which the executable runs, unless that component itself accompanies
the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot
use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined
library, provided that the separate distribution of the work based on
the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies,
or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify

or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS"

WITHOUT WARRANTY OF ANY
KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE
LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME
THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN
WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY
AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU
FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR
CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE
LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING
RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A
FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF
SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH
DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that
everyone can redistribute and change. You can do so by permitting
redistribution under these terms (or, alternatively, under the terms of the
ordinary General Public License).

To apply these terms, attach the following notices to the library. It is
safest to attach them to the start of each source file to most effectively
convey the exclusion of warranty; and each file should have at least the
"copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or
modify it under the terms of the GNU Lesser General Public
License as published by the Free Software Foundation; either
version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public
License along with this library; if not, write to the Free Software
Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the
library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!